INTERACTIVE BAYESIAN ANALYSIS USING ACCURATE ASYMPTOTIC APPROXIMATIONS

by

Luke Tierney, University of Minnesota
Robert E. Kass, Carnegie Mellon University
and
Joseph B. Kadane, Carnegie Mellon University

University of Minnesota
Technical Report No. 487
March 1987

1. Introduction

A major impediment to the use of Bayesian methods in practice is the lack of adequate software. No statistical software system in widespread use today supports Bayesian analyses. Performing such an analysis would therefore require writing a considerable amount of specialized computer code to carry out the necessary calculations. Since the time required by this programming effort would reduce the time available for considering the statistical aspects of a problem this makes a Bayesian approach considerably less attractive than it might be otherwise. An objective of the research reported in this paper is to remedy this situation by developing tools that will facilitate the routine use of Bayesian methods.

We begin by imagining a typical analysis problem in which an analyst or a team of analysts might have, say, four or five person days available to devote to an analysis. As much of this time as possible should be spent on the statistical aspects of the problem rather than on computational aspects. It should be possible, at a given stage in the analysis, to formulate an appropriate model and prior distribution, communicate this formulation to a computing system, and obtain answers, or at least reasonable approximations to answers, based on this formulation. The computational parts of this process should consume as little time and effort as possible.

This description can be viewed as a loose specification of the requirements for a software system to support Bayesian computing. The system must be flexible enough to handle formulations that are appropriate for a wide variety of problems, and will usually not fall into a convenient conjugate prior-likelihood combination. Computations must be done using algorithms capable of producing reasonable answers in a reasonable amount of time with a minimum of intervention. Finally, the system should make it as easy as possible for the user to communicate the problem formulation to the computer and analyze the results of the computations. In Bayesian analyses these results will often be graphical in nature, consisting for example of plots of posterior distributions.

To produce a satisfactory environment for supporting Bayesian analyses we thus

require adequate computational algorithms for evaluating, say, posterior moments and marginal densities, as well as a supporting shell that facilitates the application of these algorithms to a particular problem. In this paper we will describe a preliminary implementation of such an environment. The algorithms we use for the basic computations are based on recently developed second order approximation methods. These will be discussed briefly in the following section. The shell is provided by the S software system developed at Bell Laboratories. The interface between the algorithms and S will be described in Section 3. To illustrate the use of this environment we present a brief example in section 4, and we conclude in section 5 with a discussion of some of the limitations of the present framework and possible directions for further development.

2. Asymptotic Approximations

In this section we will discuss methods for approximating posterior moments and marginal densities. These calculations require a method for evaluating certain integrals. One possible approach is to use numerical integration methods, such as Gauss-Hermite quadrature (see Naylor and Smith 1982 or Smith et al. 1985) or Monte-Carlo integration (see Kloek and Van Dijk 1978, Zellner and Rossi 1984, or Geweke 1986). Both approaches are reasonable in many cases. However, in many problems, especially those with four or more parameters, these approaches will require a considerable amount of computing time, making them somewhat less attractive for preliminary stages of an analysis in which it would be useful, for example, to be able to compare posterior means calculated under several different assumptions. For these preliminary stages we prefer to use approximations that are somewhat less accurate but can be computed more rapidly, reserving slower, more accurate approximation methods for later stages of an analysis.

Posterior distributions are often approximated by a normal distribution with mean equal to the maximum likelihood estimator and covariance matrix equal to the observed information matrix. Often this approximation will suffice, but in many cases it will not. In particular, it does not capture any possible skewness in the posterior, nor is it

able to reflect the influence of the prior on the posterior distribution.

To overcome these inadequacies one can compute second order approximations. Several authors have investigated such approximations, including Lindley (1980), Mosteller and Wallace (1984), Johnson (1967), Leonard (1982), and Tierney and Kadane (1986), among others. All are based on the use of Taylor series expansions of the integrands of the integrals to be computed, though the details of the expansions vary somewhat. The approximation proposed by Lindley for example requires the explicit computation of third derivatives of the log posterior density for evaluating posterior means; fourth and fifth derivatives are required for posterior variances. The approach in Tierney and Kadane avoids the need for calculating more than second order derivatives by using different centers for the Taylor expansions of the numerator and the denominator integrands.

In our implementation described below we have chosen to use the second order approximations of Tierney and Kadane as the default method for computing posterior means and standard deviations. These approximations apply directly the computation of posterior means and variances of a positive function of the parameters. Several alternatives are available for adapting the approximations to functions that take on both positive and negative values. One approach, the approach used in the implementation described below, is to approximate the moment generating function of the posterior distribution of the function as described above, and then numerically differentiate the approximate moment generating function. A second alternative is to add a large constant to the function, thus making it essentially positive. It can be shown that both approaches are formally equivalent and, for approximating the mean, are formally equivalent to Lindley's approximation. Further discussion of this point can be found in Tierney, Kass and Kadane (1986).

Laplace's method for integrals can also be used to integrate out all but one or two components of a vector of parameters from a joint posterior density to produce an approximate marginal posterior density. This approach, proposed by Leonard (1982) and Phillips (1983), leads to an approximation that is more accurate than Edgeworth-type expansions about the posterior mode. The performance of these approximations is very

similar to the performance of the saddle point approximation for sampling distributions as discussed for example in Daniels (1954) and Barndorff-Nielsen and Cox (1979). Further details on the approximation of marginal posterior densities are given in Tierney and Kadane (1986) and Kass, Tierney and Kadane (1987)

3. Implementation

Our implementation consists of two parts: A set of Fortran subroutines for computing the basic approximations, and an interface to the S system. S, described in Becker and Chambers (1984), is a system for interactive data analysis and graphics. Its top level is written as a language in which the user can apply functions to data sets, write loops for repeated application of functions, etc.. S is designed to allow users to add their own functions (see Becker and Chambers (1985)), which is the approach we have used in providing an interface to our basic Fortran routines.

There are several reasons why we chose to work within S. One reason is the ease of adding functions to the basic system. Another is the flexibility of S itself. S can be used to perform any necessary preprocessing of input data and postprocessing of results. A single command line can be used to compute an approximation to a marginal posterior density at a specified set of points and pass the result on to an S plotting function. Furthermore, since S provides looping and control structures a user can, for example, write a short set of S statements to compute approximate posterior means for a set of data with each observation deleted, one at a time. Thus it is not necessary to select a particular form of sensitivity analysis and hard code it into the Fortran library. Instead a user can decide what form of analysis is appropriate in a given problem and carry out the computations in S itself. If a particular approach is found to be of general use it can be coded into an S macro.

The Fortran subroutines could also be used as a stand-alone library. However a user would then have to prepare a driver program for preprocessing any data that is to be analyzed, calling the approximation routines, and processing the results.

Our implementation consists of five S functions:

peval - evaluates the log posterior density (mostly used for debugging)

pmode - computes the mode of the joint posterior density (provides an interface to the HYBRJ1 routine in the MINPACK library)

pmoment - computes approximate posterior means and variances

pmar1 - computes approximate one dimensional marginal posterior densities

pmar2 - computes approximate two dimensional marginal posterior densities

To use these functions the user must supply a Fortran or C subroutine for evaluating the log posterior density and any functions of the parameters, other than coordinate functions, that might be of interest. The subroutine may also evaluate first and second derivatives of the log posterior density; if it does not then these will be evaluated numerically. In addition the user must supply an S data structure containing certain information needed by the approximation functions, including the name of the file containing the subroutines, the number of derivatives computed in this subroutine, an initial guess for the posterior mode, and an initial guess for the posterior standard deviations, which is used in evaluating numerical derivatives and for scaling certain matrix computations. This structure can also contain the data to be used and a vector of hyperparameters for the prior or the likelihood. Including these in the structure simplifies the commands required to invoke these functions.

We conclude this section with a brief discussion of some details of the implementation of these functions. A reader interested primarily in their use might wish to proceed directly to the example discussed in the next section.

A major difficulty in providing a flexible system for Bayesian analysis is that the user must be able to specify a function, a posterior density or a prior - likelihood pair, and that the evaluation of this function must be as fast as possible, since it is required many times in the innermost computational loop. In this respect the requirements for a

flexible Bayesian system are quite similar to the requirements for any system for the analysis of nonlinear regression models.

We considered several approaches to this problem. One possible approach would have been to allow the user to specify an S expression for evaluating the log posterior density. This would have been preferable from the user's point of view, but would have required either a mechanism for calling S from within a Fortran routine to obtain the value of the density at a particular point or the recoding of the approximations as S macros. Because of the interpretive nature of these approaches we believe that both would have led to unacceptably slow performance on the hardware available and we therefore decided to require the user to specify the log posterior density in a lower level language, such as Fortran, that can be compiled into machine code. An additional consideration is that the current version of S only supports single precision arithmetic, which is not accurate enough for the approximations we used.

Even the use of Fortran subroutines leads to some interesting issues, given the current S implementation in a UNIX environment: The user's subroutine code has to be linked into the S functions in some way. The approach we decided to use is to have the posterior subroutines compiled into a stand-alone program. When one of the approximation functions, such as pmoment, is invoked it will locate the appropriate files, compile the files if the source files have been modified more recently than the object files, and fork off a process to run the posterior evaluator. The two processes communicate through UNIX pipes, with arguments being sent down one pipe and results returned up another. This is similar to the present implementation of device drivers in S. It is worth noting that this approach can also be used to implement an S function for least squares fitting of nonlinear regression models. In particular, as a preliminary experiment we implemented an S interface to the nl2sn routine of Dennis, Gay and Welsch (1981) by this approach. This approach is preferable to our first approach in which we would recompile the S functions for each new model. This is a slow process, which thus discourages a user from switching between several alternative models in an analysis. In addition, it requires a significant amount of disk storage space and makes it

difficult to develop a library of standard models.

4. An Example

To illustrate the use of the S functions described in the previous section we will demonstrate their use in a problem examined by Kadane (1985). The data are taken from the National Crime Survey and consist of interviews of households taken six months apart. In each interview a member of a household is asked whether anyone in the household has been the victim of a crime during the preceding six month period. The primary question of interest is whether there is any association between victimization in the two periods. Unfortunately, a significant fraction of the interviews could not be completed, resulting in partially or completely missing information on some households. The data are given in the following table.

Table 1
Victimization Results from the National Crime Survey

	2nd visit		
1st visit	Crime-free	Victims	Non-Response
Crime-free	392	55	33
Victims	76	38	9
Non-Response	31	7	115

If there had been no missing data we would have had a 2×2 contingency table with, say, u_1 the probability that a household is crime-free in both periods, u_2 the probability that it is crime-free in period 1 and victimized in period 2, u_3 that it is victimized in period 1 and crime-free in period 2, and $u_4 = 1 - u_1 - u_2 - u_3$ that it is victimized in both periods. The likelihood function would have been

$$u_1^{n_1} u_2^{n_2} u_3^{n_3} u_4^{n_4},$$

with $n_1 = 392$, $n_2 = 55$, $n_3 = 76$, and $n_4 = 38$. A convenient choice of prior for a

likelihood of this form is a Dirichlet distribution with density proportional to

$$u_1^{b_1-1}u_2^{b_2-1}u_3^{b_3-1}u_4^{b_4-1}$$

for some $\mathbf{b} = (b_1, b_2, b_3, b_4)$. Kadane considers two "noninformative" Dirichlet prior distributions, the Haldane prior with $\mathbf{b} = (0, 0, 0, 0)$ and the Jeffreys prior with $\mathbf{b} = (.5, .5, .5, .5)$, as well as an informative prior with $\mathbf{b} = (7.5, 1, 1, 0.5)$. He uses the odds ratio

$$\psi = u_1u_4/(u_2u_3)$$

as a measure of association, with $\psi > 1$ representing positive association.

The presence of missing data complicates the problem. As a first approach we might assume that the data are missing at random, or that the mechanism generating missing observations is ignorable. Under this assumption we can ignore the cell corresponding to missing information in both periods and can view the two margins with information missing from only one period as supplemental samples from two binomial populations with success probabilities $u_1 + u_3$ and $u_2 + u_4$, respectively. This produces a likelihood of the form

$$u_1^{n_1}u_2^{n_2}u_3^{n_3}u_4^{n_4}\,(u_1+u_2)^{n_{12}}(u_3+u_4)^{n_{34}}(u_1+u_3)^{n_{13}}(u_2+u_4)^{n_{24}},$$

with $n_{12} = 33$, $n_{34} = 9$, $n_{13} = 31$ and $n_{24} = 8$. Finally, we might wish to be able to examine the implications of this "missing at random" assumption. A first approach might be to define the probabilities

$$\alpha = P\{\text{non-response} \mid \text{victimized}\}$$

and

$$\beta = P\{\text{non-response} \mid \text{crime-free}\},$$

assume that α and β are the same for both periods and that response decisions are conditionally independent given the victimization status. The data may then be viewed as arising from a nine cell multinomial distribution. If α and β are regarded as known then the likelihood can be shown to depend on α and β only through their ratio $\gamma = \alpha/\beta$. In particular, it is proportional to

$$u_1^{n_1} u_2^{n_2} u_3^{n_3} u_4^{n_4} (u_1+\gamma u_2)^{n_{12}} (u_3+\gamma u_4)^{n_{34}} (u_1+\gamma u_3)^{n_{13}} (u_2+\gamma u_4)^{n_{24}} \times (u_1+\gamma u_2+\gamma u_3+\gamma^2 u_4)^{n_{1234}},$$

with $n_{1234} = 115$. The missing at random model then corresponds to $\gamma = 1$.

To examine this problem using our S functions we first need to write a short Fortran program to evaluate the log posterior as a function of u_1, u_2 and u_3, and a subroutine to evaluate two additional functions of the parameters, $u_4 = 1 - u_1 - u_2 - u_3$ and ψ. The code is given in Listing 1.

Listing 1: Fortran Code for Victimization Example

```
      subroutine llike(n, theta, rows, cols, datmat,
     &     numhyp, hyper, fval)
      double precision theta(1), datmat(1),
     &     hyper(1), fval
      integer n, rows, cols, numhyp
      external f0
      call f0(n, theta, datmat(1), datmat(2),
     &     datmat(3), datmat(4), datmat(5), datmat(6),
     &     datmat(7), datmat(8), datmat(9),
     &     hyper(1), hyper(2), hyper(3), hyper(4),
     &     hyper(5), fval)
      return
      end
C
C     log joint posterior density
C
      subroutine f0(n,theta,n1,n2,n3,n4,
     &  n12,n34,n13,n24,n1234,b1,b2,
```

```
     &  b3,b4,gamma, fval)
      implicit real*8(a-h,o-z)
      dimension theta(3)
      double precision n1,n2,n3,n4,
     &  n12,n34,n13,n24,n1234,
     &  b1,b2,b3,b4,gamma
      double precision fval
      u1=theta(1)
      u2=theta(2)
      u3=theta(3)
      u4=max(1.0-u1-u2-u3,0.0)
      temp=(n1+b1-1.0)*log(u1)+(n2+b2-1.0)*log(u2)
     &   +(n3+b3-1.0)*log(u3)+(n4+b4-1.0)*log(u4)
     &   +n12*log(u1/gamma+u2)+n34*log(u3/gamma+u4)
     &   +n13*log(u1/gamma+u3)+n24*log(u2/gamma+u4)
     &   +n1234*log(u1/gamma**2
     &   +u2/gamma+u3/gamma+u4)
      fval=temp
      return
      end
C
C     parameter functions - u4 and odds ratio
C
      subroutine parfns(n,k,x,rows,cols,data,numhyp,
     &     hypers,gval)
      double precision x(n), data(rows,cols),
     &     hypers(numhyp),gval
      integer n, k, rows, cols, numhyp
      double precision u1, u2, u3, u4
      u1=x(1)
      u2=x(2)
      u3=x(3)
      u4=max(1.0-u1-u2-u3,0.0)
      if (k.eq.1) then
        gval=u4
      else if (abs(u2*u3).gt..0000001) then
        gval=u1*u4/(u2*u3)
      else
        gval=0.0
        end if
      return
      end
```

```
C
C     log prior - returns zero since llike computes
C     log joint posterior
C
      subroutine lprior(n,theta,numhyp,hyper,fval)
      double precision theta(1), hyper(1), fval
      integer n, numhyp
      fval=0.0
      return
      end
```

Next, we need to prepare an S data structure containing information to be used by the posterior approximation functions. The data structure for this problem is called "missing" and is shown in Listing 2.

Listing 2: S data structure "missing"

```
$file
    "missing"
$derivs
     0   0   0
$scale
    0.02  0.01  0.01
$hypers
     0   0   0   0   1
$data
    392   55   76   38   33    9   31    8  115
$mode
    0.698  0.098  0.135
$datacols
     1
$numhypers
     5
$numfunctions
     2
$numparams
     3
```

We are now ready to proceed with the analysis within S. The first step might be to compute the joint mode of the posterior. The initial guess used in the structure "missing" is the maximum likelihood estimate based on ignoring the missing data. The command for computing the posterior mode and its result are

```
> pmode(missing)
    0.6989240  0.0984498  0.1347950
```

These are the modal values for u_1, u_2 and u_3. Next, we compute the means and standard deviations of parameters with indices 1 through 5 (1 through 3 are u_1, u_2 and u_3, 4 is u_4 and 5 is ψ):

```
> pmoment(missing, 1:5)
$mean
    0.6959643  0.0994577  0.1355431  0.0690348  3.678136
$stdev
    0.0186863  0.0124064  0.0140862  0.0104993  0.9180747
```

These calculations are based on the hyperparameter vector included in the "missing" structure, which corresponds to the Haldane prior distribution and the "missing at random" assumption $\gamma = 1$. To examine the effect of varying the prior we supply an alternate hyperparameter vector to the pmoment function. This vector overrides the vector in the "missing" structure. The first four entries in this vector are the elements of the parameter vector $\mathbf{b}$ of the Dirichlet prior distribution, and the fifth entry is the value of γ. The moments for the Jeffreys prior and Kadane's informative prior are:

```
> pmoment(missing, 1:5, hypers=c(.5,.5,.5,.5,1))
$mean
    0.6945084  0.0999542  0.1359121  0.0696255  3.672527
$stdev
    0.0186998  0.0124200  0.0140887  0.0105564  0.9127732

> pmoment(missing, 1:5, hypers=c(7.5,1,1,.5,1))
$mean
    0.6968355  0.0994736  0.1349625  0.0687286  3.679743
$stdev
    0.0184832  0.0123067  0.0139462  0.0104073  0.9108143
```

It appears that the particular prior distribution used does not have much effect on the results obtained. This is not surprising in view of the total sample size in this problem. Note that all three prior distributions produce a posterior mean of around 3.5 and a posterior standard deviation of around 1 for the odds ratio ψ.

To examine the sensitivity of our conclusions to the "missing at random" assumption we examine the marginal posterior density of ψ for Kadane's informative prior and several choices of the parameter γ. The sequence of commands

```
> plot(?normalize(pmar1(missing,5,seq(1,10,len=50),
+ hypers=c(7.5,1,1,.5,1))),type="l")
> lines(?normalize(pmar1(missing,5,seq(1,10,len=50),
+ hypers=c(7.5,1,1,.5,2))))
> lines(?normalize(pmar1(missing,5,seq(1,10,len=50),
+ hypers=c(7.5,1,1,.5,10))))
> lines(?normalize(pmar1(missing,5,seq(1,10,len=50),
+ hypers=c(7.5,1,1,.5,.1)))),
```

together with some additional commands for labeling, produce the plot shown in Figure 1. The curve for $\gamma = 10$ is close to the curve for $\gamma = \infty$ in which all missing observations are assumed to correspond to victimizations. The case $\gamma = .1$ on the

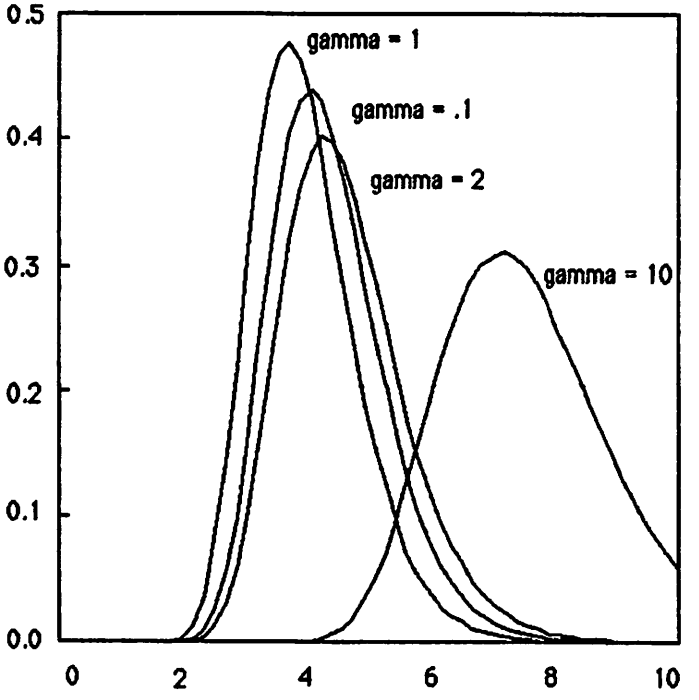

Figure 1: Posterior Density of the Odds Ratio

other hand is quite close to $\gamma = 0$ in which all missing observations are assumed to be crime-free. It seems plausible to us that γ should be greater than one, but not by a factor of 10; a value of two to three would seem to be more realistic. If γ is in fact in the range between 2 and 3 then this would not appear to produce conclusions that are substantially different from the conclusions under the "missing at random" assumption. A much larger value of γ would, however, shift the posterior on ψ considerably. At this point we could decide to pursue this issue further, perhaps by including α and β as parameters in our model. On the other hand, if we were primarily interested in determining whether ψ is greater than one then the results in Figure 1 are quite conclusive: No matter what value of γ is used, all posterior densities assign essentially

probability one to the event $\{\psi > 1\}$; in fact, they assign at least probability .9 to the event $\{\psi > 2\}$.

5. Conclusions

A considerable amount of work still remains to be done on developing and understanding approximations for use in Bayesian analyses. Diagnostics for determining when approximations are performing satisfactorily or when transformations of parameters might produce improved approximations are needed. More research is also needed on the use of numerical derivatives in approximations. Practical experience with numerical derivatives has been favorable so far, but further work is needed to determine the best form of numerical derivatives to use and to produce guidelines on when they can be used with confidence. Since manual coding of analytical formulas for derivatives is a tedious and error prone process, the ability to use either numerical derivatives or an automated system for producing analytic derivative expressions (i. e. a symbolic differentiation system) is essential for an effective computational system.

The implementation of the approximations within S has provided a reasonable framework for routine Bayesian analysis. One addition to this framework that would be desirable and will be explored in the near future is an option to carry out a more careful evaluation of a posterior moment or marginal distribution using either Monte-Carlo integration or Gauss-Hermite quadrature. A user might then begin an analysis using quick approximations to explore several different approaches, and then use slower, more accurate methods on a particular formulation that seems promising.

The weakest point in the present implementation, from the point of view of a user, is the need to prepare a Fortran subroutine for evaluating the log posterior density. It is possible to alleviate this to some extent by developing libraries of subroutines for standard models and augmenting these as needed. However the process of adding a new model tailored to a particular application is still rather onerous. Even a very simple Fortran program such as the one given in Listing 1 above can take several hours to write and debug. On the other hand, an S expression for evaluating a log posterior can usually

be written and debugged with considerably less effort. For the example of the previous section this should take on the order of five to ten minutes. To be completely satisfactory our system should permit us to write expressions for the required functions in S rather than Fortran or C. Developments in the S system discussed in Chambers (1987) and Bates and Chambers (1987) should make this possible. The performance penalty may, however, be severe since expressions written in S would have to be interpreted by S. If this is the case then perhaps we will need to explore the use of alternative high - level languages that provide facilities for compilation into machine code. A possibility, advocated for example by McDonald and Pedersen (1986), is the development of a statistical system built on a Lisp environment.

Acknowledgements

We would like to thank Michael Meyer for valuable conversations on the implementation of our S interface and Kathryn Chaloner for helpful comments on an earlier draft of this paper.

References

Barndorff-Nielsen, O. and D. R. Cox, (1979), "Edgeworth and saddle point expansions with statistical applications," J. Roy. Statist. Soc. Ser. B, vol. 41, pp. 279-312.

Bates, D. M. and J. M. Chambers, (1987) "Statistical Models as Data Structures," this volume.

Becker, R. A. and J. M. Chambers, (1984), "S - An Interactive Environment for Data Analysis and Graphics," Wadsworth, Belmont, Ca.

Becker, R. A. and J. M. Chambers, (1985), "Extending the S System," Wadsworth, Belmont, Ca.

Chambers, J. M. (1987), "Interfaces to a Quantitative Programming Environment", this volume.

Daniels, H. E., (1954), "Saddle point approximations in statistics," Ann. Math. Statist., vol. 25, pp. 632-650.

Dennis, J.E., D.M. Gay, and R.E. Welsch, (1981), "An Adaptive Nonlinear Least-Squares Algorithm," Acm Trans. Math. Software, vol. 7, pp. 369-383..

Geweke, John, (1986), "Bayesian inference in econometric models using Monte Carlo integration," Preliminary Report, Department on Economics, Duke University.

Johnson, R. A., (1967), "An asymptotic expansion for posterior distributions," Ann. Math. Statist., vol. 38, pp. 1899-1907.

Kadane, J. B. (1985), "Is Victimization Chronic? A Bayesian Analysis of Multinomial Missing Data," J. of Econometrics, vol. 29, pp. 47-66.

Kass, Robert E., Luke Tierney, and Joseph B. Kadane, (1986), "The validity of stochastic asymptotic expansions based on Laplace's method," Technical Report, Department of Statistics, Carnegie Mellon University.

Kass, R. E., L. Tierney, and J. B. Kadane, (1987), "Asymptotics in Bayesian Computation," to appear in Bayesian Statistics 3, ed. J. M. Bernardo, M. H. De Groot, D. V. Lindley and A. F. M. Smith.

Kloek, T. and H. K. Van Dijk, (1978), "Bayesian estimation of equation system

parameters: an application of integrat ion by Monte Carlo," Econometrica, vol. 46, pp. 1-19.

Leonard, T., (1982), "Comment on "A simple predictive density function"," J. Amer. Statist. Assoc., vol. 77, pp. 657-658.

Lindley, D. V., (1980), "Approximate Bayesian methods," in Bayesian Statistics, ed. J. M. Bernardo, M. H. De Groot, D. V. Lindley and A. F. M. Smith, University Press, Valencia, Spain.

McDonald, J. A. and J. Pedersen, (1986), "Computing Environments for Data Analysis, Part 3: Programming Environments," Technical Report No. 82, Department of Statistics, University of Washington.

Mosteller, F. and D. L. Wallace, (1984), "Applied Bayesian and Classical Inference: The Case of the Federalist Papers," Springer, New York.

Naylor, J. C. and A. F. M. Smith, (1982), "Applications of a method for the efficient computation of posterior distributions," Applied Statistics, vol. 31, pp. 214-225.

Phillips, P. C. B., (1983), "Marginal densities of instrumental variable estimators in the general single equation case," Advances in Econometrics, vol. 2, pp. 1-24.

Smith, A. F. M., A. M. Skene, J. E. H. Shaw, J. C. Naylor, and M. Dansfield, (1985), "The implementation of the Bayesian paradigm," Commun. Statist., vol. 14, pp. 1079-1092.

Tierney, L. and J. B. Kadane, (1986), "Accurate approximations for posterior moments and marginal densities," J. Amer. Statist. Assoc., vol. 81, pp. 82-86.

Tierney, L. , R. E. Kass, and J. B. Kadane, (1986), "Approximation of posterior expectations and variances using Laplace's method," Technical Report, Department of Statistics, Carnegie Mellon University.

Zellner, A. and P. E. Rossi, (1984), "Bayesian analysis of dichotomous quantal response models," J. of Econometrics, vol. 25, pp. 365-393.